



NOTE:This version was created in error and never launched. Leave this version as "Unpublished" until an actual version is required and paste over this one.

Introduction

Overview

Welcome! This document outlines the sequence of steps required to develop the necessary competence in introductory programming using Java. To pass this assessment you will have to successfully develop two applications. It is not recommended that you pick up a text on Java and abandon the activities outlined in this course of study. This course of study outlines the essentials you must know and the programming activities selected are relevant to the performance tasks.

Passing the performance assessment requires successful completion of two performance tasks in TaskStream.

- For the first task, you will be expected to develop an application that has a graphical user interface (GUI). The application will require you to develop algorithms to perform the calculations required as well as display results.
- For the second task, you will be expected to create an application that performs text file processing. Your application will read in one or more files, perform the required processing or analysis, and generate the appropriate output files.

You need to be able to complete the activities provided by this course of study to be sure you are ready for the performance assessment.

Outcomes and Evaluation

There are 3 competencies covered by this course of study; they are listed in the "[Competencies for Introduction to Programming \(AAT1\)](#)" page.

Performance Assessment

You will complete the following performance assessment in [TaskStream](#) :

- AAT1

Previews of task instructions and evaluation rubrics for this assessment are available via the "Assessment Preparation" box in the online course of study.

Preparing for Success

The information in this section is provided to help you become ready to complete this course of study. As you proceed, you will need to be organized in your studies in order to gain competency in the indicated areas and prepare yourself to pass the final assessments.



Your Learning Resources

Enroll in or order the learning resources for this course as early as possible so as to give them time to arrive and give you enough time to become familiar with them.

Automatically Enrolled Learning Resources

You will be automatically enrolled at the activity level for the following learning resources. Simply click on the links provided in the activities to access the learning materials.

WileyPLUS: Java for Everyone

The following textbook is available to you as an e-text within this course of study. You will be directly linked to the specific readings required within the activities that follow.

- Horstmann, C.S. (2010). *Java for Everyone*. New York.: Wiley. ISBN-13: 978-0471791911

WileyPLUS is an online learning environment that includes:

- the entire e-text for the *Java for Everyone* book,
- programming activities,
- source code for chapter examples,
- animations that demonstrate chapter topics, and
- screencasts.

For each WileyPLUS chapter, complete the following:

1. **Readings:** full text for the chapter
2. **Animations:** demonstrations of chapter's key concepts
3. **Screencast:** instructional videos
4. **Practice:** drill and practice quizzes (optional) and assignments.
5. **Instructor-provided materials:** The instructor-provided activities include a list of selected review and programming exercises that are directly relevant to the competencies you need.

Please keep your own copy of these out when you start a new chapter so you know which review and programming exercises you need to do and which you can skip. This will expedite your preparation.

Additional Preparation

There are many different learning tools available to you within your course of study in addition to the learning resources already discussed. Take the time to familiarize yourself with them and determine how best to fit them into your learning process.

Activities at a Glance Outline



This course of study includes an "[Activities at a Glance](#)" outline to help you briefly visualize the elements of this course of study.

This outline will also provide general pacing guidelines for your work through the course of study and can be printed and used as a quick checklist for your progress.

Message Boards, Study Notes, FAQs

Message boards, study notes, and FAQs are available in every course of study.

Use the "[Additional Learning Tools](#)" page to review these tools.

The WGU Central Library

The [WGU Central Library](#) is available online to WGU students 24 hours a day. The library offers access to a number of resources, including over 60,000 full-text e-books; articles from journals, magazines, and newspapers; course e-reserves; and tutorials on how to use these resources and the library. The library also includes a reference service for help with research questions or navigating the library.

For more information about using the WGU Library, view the "WGU Library: Finding Articles, Books & E-Reserves" video in the Student Resources section of [The WGU Channel](#).

Course Mentor Assistance

Course mentors are available to help you. Their job is to aid understanding in areas where you need to improve and to guide you to learning resources. Request their help as needed when preparing for assessments.

Course mentors cannot provide reviews of entire assessments. If you fail assessment attempts, review the provided feedback first, then ask the course mentor specific questions about what you can do to meet the competency standard. Request course mentor assistance as necessary in preparing for second attempts at objective assessments or performance task revisions. Mentors cannot guarantee you pass as they do not evaluate assessments; however, they can provide the assistance and advice necessary to help you succeed.

SkillSoft Mentoring

In addition to your course mentor, SkillSoft mentors are available to assist students with Java related questions. If you have a question about any of the Java concepts, a SkillSoft mentor can explain the concept and help you until you have a clear understanding.

You can contact these mentors by chat or email via the [SkillSoft "Ask My Mentor" site](#) :

Note: You will need to select "Play this course" in the window that pops up to be able to enter the [SkillSoft "Ask My Mentor" site](#).



- **Chat with a Mentor:** Available Monday-Friday 9-5 Eastern Time. There is an option to email the chat transcript when the chat concludes.
- **Email My Mentor:** Email turnaround time is 6 hours if sent during normal business hours (Monday-Friday 9-5 Eastern Time). Emails sent outside of business hours will be responded to on the next business day.

Other Preparations

You will need to download other learning resources for this course of study:

NetBeans and/or the Java SE Development Kit (JDK)

You will be required to develop Java applications for this course of study, Introduction to Programming.

NetBeans is the recommended integrated development environment (IDE). This IDE will provide you an editor, debugger, and the Java Development Kit.

Note: If you choose NetBeans as your IDE, you will not have to download the Java SE Development Kit (SDK) as a separate activity. You can download the SDK bundled with NetBeans (recommended).

Download this resource from the following webpage. Then follow the installation instructions for Java SE NetBeans IDE Download Bundle.

- "[Java SE Downloads](#)"

Demonstration Screenshots in NetBeans

Download the following document:

- "[Demonstrations of First 4 Activities with Screenshots](#)".

This document provides screenshots of step-by-step demonstrations of completing the activities in the "Editing, Compiling, Running, and Debugging in NetBeans" topic section (within the "Getting Familiar with the Integrated Development Environment" subject section) of the course of study.

Use this document to help guide you as you complete these activities on your own. These activities will familiarize you with your development environment.

Getting Familiar With the Integrated Development Environment

You will learn how to create a new Java project in NetBeans and also get familiar with using the debugger. The NetBeans website has some excellent tutorials to help you get started.



As you complete the activities in this section, refer to the "[Demonstrations of First 4 Activities with Screenshots](#)" document that you downloaded in the "Preparing for Success" section.

Editing, Compiling, and Debugging

There are three activities necessary to develop working applications:

- editing,
- compiling, and
- debugging.

To write your applications, you need an editor. Running an application requires compilation. During testing, you may realize the results are not what you expected and software errors (bugs) are present.

In this topic, you will learn how to use tools provided by NetBeans that allow you to edit, compile, and debug your application.

NetBeans Development Tools for Editing, Compiling, and Running

To get familiar with the NetBeans IDE, complete the activities in the following document:

- "[Introduction to the IDE](#)"

Basic Application Development

You will begin your Java journey by learning the fundamentals of Java programming and the basic syntax for Java. You will learn how to create algorithms. An algorithm is a series of steps that will solve a problem.

Outlining the steps required to meet an application's requirements will help you create an accurate design before you code. This process will result in less time spent troubleshooting logical errors in your code. Knowing basic syntax is important to developing applications that will compile and run. The syntax has to be exact or the application will not compile.

Turning Algorithms Into Java Applications

Turning Algorithms Into Java Applications Do you think Java is interpreted or compiled?

- How are command line arguments passed into a Java application?

WileyPLUS Chapter 1: Introduction

Read the following chapter in *Java for Everyone*:



- [chapter 1 \("Introduction"\)](#)

Pay close attention to section 1.7 ("Algorithms"). This section provides you with an introduction to algorithm development. Most of your applications will require algorithm development.

Algorithm development is very important and requires practice.

Please complete the following Review Exercises in [chapter 1 \("Introduction"\)](#) of *Java for Everyone*:

- R1.6
- R1.7
- R1.11
- R1.13
- R1.14
- R1.18

Now, complete the following Programming Exercises in [chapter 1 \("Introduction"\)](#) of *Java for Everyone*:

- P1.1
- P1.8
- P1.10

Variables, Data Types, and Strings

Variables and data are a part of most applications and certainly a part of the applications you will need to write for your performance assessment.

This topic introduces you to variables and the data types provided by Java.

WileyPLUS Read, Study, and Practice

Read the following chapter in *Java for Everyone*:

- [chapter 2 \("Fundamental Data Types"\)](#)

You have to apply what you are learning. Typing out code helps strengthen your ability to apply these principles and get the Java syntax down.

These exercises will give you practice with creating Java arithmetic expressions, creating formatted output, manipulating Strings, and working with variables. It is very important that you complete each exercise.

Please complete the following Review Exercises in [chapter 2 \("Fundamental Data Types"\)](#) of *Java for Everyone*:



- R2.1
- R2.3
- R2.4
- R2.9
- R.2.14
- R.2.15

Now complete the following Programming Exercises in [chapter 2 \("Fundamental Data Types"\)](#) of *Java for Everyone*.

- P2.1
- P2.2
- P2.4
- P2.5
- P2.8
- P2.9
- P2.12
- P2.13
- P2.14
- P2.19

Writing Programs That Process User Input

Access and view the video demonstrations for this chapter. These videos discuss how to get user input and use this entered data in your application.

Now, complete the activities found in the following document:

- "[User Input](#)"

Control Structures

This subject section will give you practice with using control structures to create execution paths. The majority of the applications you will develop will require the use of control structures to develop the logical flow of control of events, behavior, and actions.

Decision Structures

The purpose of conditional statements is to execute a block of code based on whether a certain condition is true or false.

If true, the block of code is executed; if false, the block of code is ignored. Decision structures are frequently required to develop program logic.

WileyPLUS Read, Study, and Practice



Read the following chapter from *Java for Everyone*:

- [chapter 3 \("Decisions"\)](#)

Notice how flowcharts are used as a visual tool to work out algorithms. It is recommended that you use flowcharts to work out your conditions until you are comfortable with applying the "if" and "if-else" control structures.

Pseudocode should always be used to work out your algorithms. Flowcharts are a great supplemental tool to help you work out how to apply a control structure within your algorithm.

Please complete the following Review Exercises in [chapter 3 \("Decisions"\)](#) of *Java for Everyone*:

- R3.2
- R3.3
- R3.4
- R3.7
- R3.8
- R3.9
- R3.10
- R3.11
- R3.13
- R3.18
- R.3.21

Now, complete the following Programming Exercises in [chapter 3 \("Decisions"\)](#) of *Java for Everyone*:

- P3.1
- P3.2
- P3.3
- P3.4
- P3.5
- P3.6
- P3.9
- P3.10
- P3.12
- P3.14
- P.3.17
- P3.23
- P3.25

Working With Others



Please go to the message board and locate a study partner or two. This activity will provide you with peers with whom you can discuss the concepts you are learning now for this course of study and also build your professional network.

Although you cannot collaborate when you complete the performance tasks, you are encouraged to work together on the practice programming activities.

Repetition and Iteration

This topic demonstrates how to use for, while, and do loops. Computer applications are often developed to handle repetitive tasks. These control structures provide the ability to repeat program statements.

Consider a task that you perform each day that is repetitive in nature. How would you write the instructions to complete this task in pseudocode?

WileyPLUS Read, Study, Practice

Read the following chapter from *Java for Everyone*:

- [chapter 4 \("Loops"\)](#)

Continue to use flowcharts to work out your conditions until you are comfortable with applying loops.

Please complete the following Review Exercises in [chapter 4 \("Loops"\)](#) of *Java for Everyone*:

- R4.1
- R4.2
- R4.3
- R4.4
- R4.5
- R4.6
- R4.7
- R4.8
- R4.9
- R4.10
- R4.11
- R4.12
- R4.13

Now, complete the following Programming Exercises in [chapter 4 \("Loops"\)](#) of *Java for Everyone*:

- P4.1
- P4.3



- P4.5
- P4.6
- P4.7
- P4.8
- P4.9
- P4.19
- P4.20

Methods

You have learned how to develop algorithms using pseudocode.

In these activities you will learn how to design and implement methods so that you are able to provide encapsulated functionality.

Design and Development of Methods

This topic demonstrates how to develop methods that will execute a sequence of instructions. You will learn how to pass parameters to and return values from methods.

WileyPLUS Read, Study, and Practice

Read the following chapter from *Java for Everyone*:

- [chapter 5 \("Methods"\)](#)

It is very important to pass parameters to methods correctly. The parameters passed to a method must match in number and type. Methods used from a class in the Java API have to be passed the correct values, which is why it is a good idea to review methods on Java API classes before you use them.

Please complete the following Review Exercises in [chapter 5 \("Methods"\)](#) of *Java for Everyone*:

- R5.1
- R5.2
- R5.3
- R5.6
- R5.8
- R5.9
- R5.10
- R5.12

Please complete the following Programming Exercises in [chapter 5 \("Methods"\)](#) of *Java for Everyone*.



- P5.1
- P5.2
- P5.3
- P5.4
- P5.6
- P5.7

Bookmark Java API Documentation

Locate the current version of the Java API Documentation online.

Note: You should consult the Java API Documentation each time you use one of the classes from the API.

Take a look around at the classes. Notice the information you can obtain for each class such as the package to which it belongs. Bookmark this link and refer to it often

Data Structures

This subject section will provide you with an introduction to data structures provided by the Java API and provide an understanding of how collections can be used.

Data structures are a fundamental computer science topic that has been studied for decades. Data structures allow you to collect a group of values or objects. Why might processing a group of values be helpful?

Arrays and ArrayLists

Collections allow you to store and manipulate a collection of objects. The Java API provides several reusable data structures that are efficiently written. Knowing how to use these collection types will save you time writing your own that will likely be less efficient.

This topic introduces you to collection types arrays and ArrayLists.

WileyPLUS Read, Study, and Practice

Read the following chapter from *Java for Everyone*:

- [chapter 6 \("Arrays and ArrayLists"\)](#)

Arrays are another type of variable you can declare. An array collects values but they have to be of the same type. You can have an array of all "ints" or an array of all doubles, for example.



Two-dimensional arrays are like a checkerboard or spreadsheet. A 2D array has rows and columns. You typically use nested "for" loops to iterate through each value of an array, whether it is for filling the array or printing its contents. You may want to review "for" loops at this time.

Please complete the following Review Exercises in [chapter 6 \("Arrays and ArrayLists"\)](#) of *Java for Everyone*:

- R6.1
- R6.2
- R6.4
- R6.7
- R6.8
- R6.11
- R6.12
- R6.13
- R6.18

Now, complete the following Programming Exercises in [chapter 6 \("Arrays and ArrayLists"\)](#) of *Java for Everyone*:

- P6.1
- P6.5
- P6.6
- P6.9
- P6.17

Readiness Check: 2D Arrays

You will need to use a 2D array in your performance assessment.

Download the following package (which includes a NetBeans application), unzip the NetBeans project, and open in NetBeans:

- "[2D Array Practice.zip](#)"

Here is an example application that creates a 2D array for the size specified by the user. The user enters the contents to be stored in the array. The contents of the array are then printed.

Complete the activities provided in the document titled "Hip Hip Array!"

Classes

Now that you have the fundamentals down concerning Java syntax and algorithm development, you are ready for the basics of object-oriented design and development. Classes are at the heart of OO design and development.



Applications developed using object-oriented principles are easier to maintain and extend their functionality. Classes contain methods and methods implement algorithms. Most software development jobs require a solid understanding of OO design and development.

Classes and Objects

Encapsulation is a key strategy for developing applications that are maintainable. Classes should be created to encapsulate data and to create code that is cohesive and easy to maintain.

When designing methods for your classes, you will want to apply your algorithm development techniques used in the earlier activities to help you develop behavior that meets requirements.

This topic provides an overview of classes, objects, attributes (instance variables), getters and setters, instance methods, and constructors.

WileyPLUS Read, Study, and Practice

Read the following chapter from WileyPLUS:

- [chapter 7 \("Objects and Classes"\)](#)

You will learn the steps for class design in this chapter.

- Why do you think it is a good idea to declare instance variables as private?
- What are the steps for class design?

Please complete the following Review Exercises in [chapter 7 \("Objects and Classes"\)](#) of *Java for Everyone*:

- R7.1
- R7.4
- R7.7
- R7.8
- R7.9
- R7.10
- R7.11
- R7.12
- R7.16
- R7.18

Now, complete the following Programming Exercises in [chapter 7 \("Objects and Classes"\)](#) of *Java for Everyone*:

- P7.6
- P7.8



- P7.10
- P7.11
- P7.13

Text File I/O and Exception Handling

Oftentimes algorithms require using data. You need a way to read and write data from sources such as files, ports, databases, the screen, and the keyboard. Input and output using the Java API I/O streams is usually a challenging subject. In this lesson you will focus on reading and writing to text files. Control structures are helpful with processing files as the processing may be repetitive.

Control structures are used to control the flow of a program when things go right and when things go wrong. In this section you are going to look at controlling flow when there is a problem at run time. This will allow you to write robust applications that can handle imperfect scenarios such as incorrect user entry or problems with the data.

Text Files

Java provides over 30 classes to handle I/O streams. This section introduces you to a few that read and write to text files, including two classes Java 5.0 introduced that make reading and writing to text files easy:

- Scanner and
- Formatter.

File input and output is one of the most difficult concepts. You will have to understand this well to complete one of the performance tasks at the end of this course of study.

You will also learn how to create your own exception handling so that your application can gracefully handle a runtime error.

WileyPLUS Read, Study, Practice

Read the following chapter from *Java for Everyone*:

- [chapter 8 \("Input/Output and Exception Handling"\)](#)

So far you have had to write applications that process text input from the keyboard. You have also written applications that streamed output to display results. You will learn how to expand streaming to include writing to and reading from files.

Please complete the following Review Exercises in [chapter 8 \("Input/Output and Exception Handling"\)](#) of *Java for Everyone*.

- R8.1



- R8.5
- R8.6
- R8.8
- R8.9

Now, complete the following Programming Exercises in [chapter 8 \("Input/Output and Exception Handling"\)](#) of *Java for Everyone*.

- P8.1
- P8.2
- P8.3
- P8.4
- P8.5
- P8.8
- P8.9
- P8.11

(Optional) Java for Everyone Chapter 9

[Chapter 9 \("Inheritance and Interfaces"\)](#) of *Java for Everyone* is not recommended reading for this course of study, but the information will be helpful for other Java-related courses at WGU. You may choose to read this, time permitting.

Collections

This lesson will provide you with an introduction to the collection hierarchy and how to collect objects in an application. Maps are also covered which store key/value pairs.

Collections and Maps

Collections allow you to store and manipulate a collection of objects. The Java API provides several reusable data structures that are efficiently written.

You became familiar with the list collection type when you learned how to use ArrayLists. You will now expand your ability to work with collections of objects using sets and maps.

WileyPLUS Read, Study, and Practice

Read the following chapter from *Java for Everyone*.

- [chapter 10 \("Collections and Maps"\)](#)

Sets allow you to collect objects that are unique. A set will politely ignore any attempts to add a duplicate.



Maps are excellent for look up tables because they pair keys with values. They are also helpful for counting as the key can represent the object you are trying to tally, and value can be the tally.

Please complete the following Review Exercises in [chapter 10 \("Collections and Maps"\)](#) of *Java for Everyone*:

- R10.1
- R10.2
- R10.16
- R10.19

Now, complete the following Programming Exercises in [chapter 10 \("Collections and Maps"\)](#) of *Java for Everyone*:

- P10.7
- P10.9
- P10.10

Algorithm Development Using a Graphical User Interface

In this lesson you will learn how to create a graphical user interface (GUI), to build applications that are user friendly. Interfaces that are intuitive, easy to use, and provide readable results positively impact satisfaction of an application.

GUI Building

In this section you will learn how to use javax.swing components to build a GUI.

This topic is typically one of the most enjoyable to learn, and you will experience the ease in which a very sophisticated user interface can be developed.

Using NetBeans to create a GUI

Any version of NetBeans will provide you with an easy and flexible way to create swing applications (GUIs). You will need to create a GUI for one of your performance tasks.

You could use a Java layout manager to layout all of your UI/swing components, but this approach is more difficult. NetBeans lets you to create Java GUI applications visually.

Please locate and complete the appropriate tutorial for your version of NetBeans from [NetBeans' website](#). Look for the Java and JavaFX GUI section of tutorials.



Starting with NetBeans 6.9, JavaFX Composer was released. This is a more advanced approach to GUI development and provides the ability to include graphics. You have the option of using JavaFX Composer, but the NetBeans GUI Builder is the easier of the two and sufficient for your performance tasks.

At a minimum, you should know how to use the following Java API classes for GUI development:

- JButton
- JLabel
- JRadioButton
- JTextField
- JTextArea

Readiness Check: Building Swing Applications

Please download the following document, which contains a link to starter code and activities:

- "[GUI/Swing Application Development](#) "

Complete the activities to practice GUI application development and to create applications that use data entered into a GUI.

Final Steps

Congratulations on completing the activities in this course of study! This section will guide you through a review and the assessment process.

Review of Major Points

Remember that control structures are vital to the development of application logic. Many applications are developed to handle repetitive tasks for which your iteration control structures will be useful.

Today's applications are typically data intensive; therefore, having solid understanding of the characteristics of common data structures and good command of the collections provided by the Java API will be important to your future application development.

Final Review

The three competencies and their associated portions of the learning resources are listed below.

Competency: Control Structures

The graduate develops working programs that use appropriate control structures and accurately evaluates paths in program code.



- *Java for Everyone*, [chapter 3\("Decisions"\)](#)
- *Java for Everyone*, [chapter 4\("Loops"\)](#)
- *Java for Everyone*, [chapter 8\("Input"\)](#)

Competency: Algorithm Design and Development

The graduate designs and develops algorithms for problem solving and implements those algorithms using appropriate program code.

- *Java for Everyone*, [chapter 1\("Introduction"\)](#)
- *Java for Everyone*, [chapter 2\("Fundamental Data Types"\)](#)
- *Java for Everyone*, [chapter 5\("Methods"\)](#)
- *Java for Everyone*, [chapter 7\("Objects and Classes"\)](#)

Competency: Use of Data Structures

The graduate develops working programs that use data structures for problem solving.

- *Java for Everyone*, [chapter 6\("Arrays and ArrayLists"\)](#)
- *Java for Everyone*, [chapter 10\("Collections and Maps"\)](#)

Assessment Information

The activities in this course of study have prepared you to complete the AAT1 performance assessment. If you have not already completed the assessment, you will do so now.

You will complete two performance tasks that will assess your ability to use the material learned in an integrated manner. This means that these tasks may require you to apply a combination of

- control structures,
- file input and output,
- GUI development,
- data structures, and
- algorithm development.

Accessing Performance Assessments

You should have completed the following tasks as you worked through this course of study. If you have not completed the tasks in [TaskStream](#), do so now.

- AAT1: Task 1
- AAT1: Task 2

For directions on how to receive access to performance assessments, see the ["Accessing Performance Assessments"](#) page.

Feedback



WGU values your input! If you have comments, concerns, or suggestions for improvement of this course, please submit your feedback using the following form:

- [Course Feedback](#)

Panopto Test 3

Testing to see if inline was the problem.

Inline test 2

<https://lrps.wgu.edu/provision/18060549>

Panopto Test 3

Testing to see if inline was the problem.

Inline test 2

<https://lrps.wgu.edu/provision/18060549>

Table Practice

Tables.

Table

Table

Column Border?	Column 2	Column 3	
Column 1	Column 2	Column 3	
Column 1	Column 2	Column 3	
Column 1	Column 2	Column 3	Column 4?
Column 1	Column 2		

Table

ADA Policy



Western Governors University recognizes and fulfills its obligations under the Americans with Disabilities Act of 1990 (ADA), the Rehabilitation Act of 1973 and similar state laws. Western Governors University is committed to provide reasonable accommodation(s) to qualified disabled learners in University programs and activities as is required by applicable law(s). ADA Support Services serves as the principal point of contact for students seeking accommodations and can be contacted at ADASupport@wgu.edu. Further information on WGU's ADA policy and process can be viewed in the student handbook at the following link:

- [Policies and Procedures for Students with Disabilities](#)